FEB/89

ZX-Appeal

vancouver sinclair users group

next meeting:

KILLARNY COMMUNITY CENTRE
6260 KILLARNY STREET
VANCOUVER

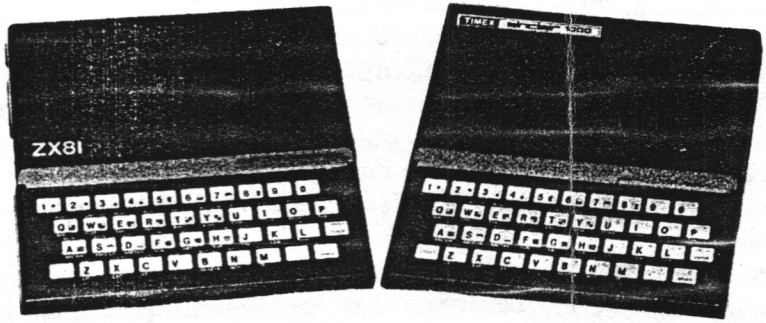
FRIDAY; 7:00PM

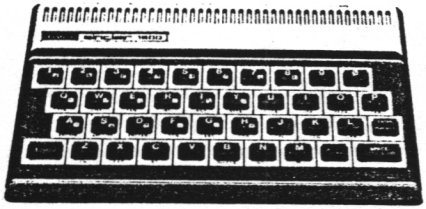
FEBRUARY 10/89

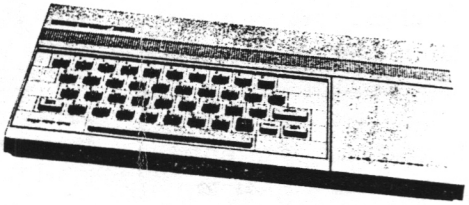
ZXAppeal is a monthly newsletter put out by the Vancouver Sinclair Users Group. For more information on the group and ZXAppeal see the backcover.

inside

THIS ISSUE..... 2
BITS&PIECES..... 2
CAPITOLFEST..... 3
MEETING DATE..... 3
MINUTES..... 4
COLOUR PLOTTING..... 5
MC POINTERS II..... 6
DISPLAY ROUTINE II..... 7
TOUCH of CLASZ..... 8
64K NVM..... 10
QL OVERHEATING..... 15
HDWR REVIEWS..... 16
PROGRAMS..... 18
CAPITOLFEST II..... 19







THIS ISSUE.....

Cold enough for you? During this cold snap why not just stay indoors, warm up your favourite soup and settle down in front of the ol' keyboard 'til the spring thaw.

This month we continue with "PART TWO" of a couple of things last month from Wilf. Fred is also back with a follow-on article to last month. Vince has been busy - he has developed the ultimate TS1000/ZX81 add-on:- an internal 64K NVM. We reprint a superb article by Pete Fischer from the December issue of TIMELINEZ - the newsletter of the Bay Area T/S Groups. (Interestingly, this newsletter has become an umbrella for not only the three Bay Area groups but also another California group as well as a group from Florida!) Pete writes about the hardware interface developed by John McMichael wherein one can interface the 2068 to the Commodore 1520 colour printer/plotter. And a few other items you might interesting.

****

BITS & PIECES.....

...why don't we set up our own BBS? I've just purchased the latest from LARKEN:-the MAXCOM Telecommunication Software Package. This program includes a 64 column Terminal Emulator and a full-feature Remote User BBS allowing Up and Downloads from an LKDOS driven system. We have enough $ in the bank to be able to afford the basic cost for a 'phone line for a few months and longer if it works out. I'm willing to use my system for the BBS and my place is already wire for a second 'phone line. All modem software, whether for the ZX81, the 2068, or the QL, could use this board for messages and 2068s could Up and Down load programs. I'm not sure but it should be possible for a ZX81 to upload and another ZX81 to download a program from disc even though it is on a 2068 system. Same goes for QLs. Let's talk about it at the meeting.

...just received the Sept/Oct issue of TIME DESIGNS. A little late but well worth the wait. Tim confirms his intentions to keep on keeping on for quite some time so let's not think negative things.

...my recent trip 'Down Under' gave me sufficient Aero Plan points for a free ticket anywhere in Air Canada's North America. So it's off to Washington, DC, the first weekend of May for the "'89 CapitolFest" , sponsored by the Capitol Area Timex Sinclair Users Group. This could well be the last of these types of affairs as our dwindling TS world won't make it worthwhiled in the future.

...want another Sinclair hi-tech, low-cost goodie? How about a Sinclair satellite dish for $300. Just released to compete with the Amstrad dish at $400. These dishes are specifically designed to receive the direct-broadcast 'Sky Television' just now becoming available in Europe.

...our support mechanisms slowly disappear:- Curry Computer, it is reported, has gone MS-DOS; both EZ-Key and FOOTE Software have sold out to RMG Enterprises.

...it is also reported elsewhere that the demise of SincWare News has been somewhat exaggerated. SWN isn't dead, it is joining with Quantum Levels magazine to form a better and stronger bi-monthly. If these guys are willing to support us then let's support them!

...the Toronto group newsletter reports there is a chap willing to perform QL repairs right here in Canada:- Schennelly Stoughton, 191 William St. Nth, Lindsay, Ont. K9V 4B8

****

RENEWING MEMBERS:

M.Vieira, H.Taylor, H.Slot, E.Sakara, K.Gamey, K.Abramson, M.Kendall, N.Trylsson, B.Watts, D.Ross, , L.Montminy, A.Boisvert

REMEMBER: If the expiry date on your envelope is high-lighted be sure to renew at the meeting or by mail.

****

1989 CapitolFest

MAY 5th, 6th &7th

at the Washington DC, Howard Johnson Plaza New Carrollton, MD. Sponsored by CATS.

Make your spring plans now!!

The Capitol Area Timex Sinclair User Group is sponsoring The 1989 Computerfest for all Timex and Sinclair Computers. These include 1000, 2068, QL, Z88 and other Sinclair related PC's. There will be a Friday night Banquet (limited seating) and 2 days of true festivities, including seminars, guest speakers, swap meet, vendors and more...

Washington holds more cultural playtime activities for your whole family than any other US city. Come and see for yourself!

To receive more information about the Fest and Washington DC, drop us a line and we'll send you an information packet:

CATS CAPITOLFEST
PO Box 24
Garrett Park, MD 20896
Contact: Audrey Curnutt
(301) 439-8756
BBS (301) 588-0579

EVENT of the YEAR!!!

****

...meeting date.

This space was reserved for the meeting minutes
but 'ye olde scribe' did not make the deadline.

Reprinted from the Dec/88 issue of TIMELINEZ

COLOR PLOTTING ON THE 2068

A HARDWARE INTERFACE FOR THE COMMODORE 1520

A review by P.Fischer

The color image you see was produced on a Commodore 1520 Printer/Plotter as interfaced to the 2068. The interface required comes from John McMichael in Wyoming. Other reviews have appeared on this system, but I always felt, "you gotta SEE it to really understand!"

This system is for people who are looking for an inexpensive way to produce hardcopy color graphics on the 2068.

The interface is, at it's minimum, a simple board that is intended to plug into an Oliger Expansion Board. But it is also easy to make it plug directly onto the computer by soldering on the appropriate connector.(or John will, for a small fee).It also comes bundled with a software patch that allows you to drive the plotter from BASIC.

There are also several additional pieces of soft- ware available to compliment this system. The two I ordered were PIC-PLOT and SCREEN-TO-PLOTTER. The former simply allows you to load a SCREEN$ and plot it in one or two colors (2 different sizes). The second allows you to control the plotter mechanism with a joystick or trackball in "real time". You can also load a SCREEN$ first, then just "trace over" that image. Using this program you can change colors whenever you want. The moves are automatically stored in RAM. When finished you can SAVE these moves to tape for later re-loading and re-plotting so you can make as many copies as you like.

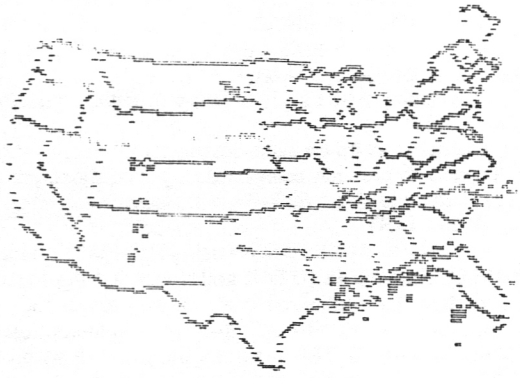
As you can see, the lines drawn by the 1520 are quite fine. This, I feel, is one of it's most impressive features. No other device for the 2068 matches this sharpness, as far as I know.

The screen you see was produced by PIC-PLOT, and this is the largest size available. There is another size, about 1/4 of this. So the 1520 obviously does not compete with a commercial plotter by any means. But the price is right; $30 for the plotter itself, and about as much for the interface. The two mentioned programs are $9 each. A recent development is that the same interface will also drive an OKIMATE 20 color printer, more on this later.

But the 1520 is a wonder unto itself. It's 6x11x 3" and the mechanism is a bit larger than a cigarette pack. John analyzed it, even went out and bought a VIC 20 to see how it was intended to work. It contains it's own ROM which gives it a larger array of functions even without the 2 special programs I got.

I used it mostly with PIC-PLOT (the easiest),you just load in a SCREEN, pick a size, pick the color or colors (red, blue, green or black) and the software does the rest.

The results are mixed. Some plots are spectacular, others only mud. But it's all worth it for the spectacular ones. After a while, you begin to have a feel for which ones most likely will work and which won't. There's a direct correlation with the complexity of the image. The more complex, the more likely the "mud" effect. Also, in the large size PIC- PLOT printout, shadows come out as white, confusing the image.



The time to plot also varies by complexity. Simple images can take as little as 7 minutes. The longest one took slightly over an hour. Obviously you can't stand there and watch it, you have to do something else while it plots.

The biggest problem with this plotter is availability. They haven't been manufactured for some time, so it's necessary to find the plotter FIRST, and THEN order the interface. Possible sources include Commodore users who are selling out their entire systems or perhaps by contacting a Commodore User Group. I bought mine, some time back, from TOYS-R-US. They at one time had 50,000 of them, but appear to have sold out. It's worth a phonecall, however, if there's a store local to you.

The second program I tested was the SCREEN-TO-PLOTTER software, and I liked it. It requires a bit of "eye-hand" coordination to work. No effortless plots here. The basic principle here is that you have a Joystick or (in my case) a trackball and the plotter running. You can move the cursor anywhere on the screen, and then hit enter the plotter will immediately draw the same line on the paper. Move the cursor again, and repeat the process. Each move is stored in ram and can be re-run for multiple copies. My first attempt was a house on a hill with the sun setting behind it. The house and hill were easy, but when I tried to draw a believeable sun, I fell on my face. I had to draw a circle on a piece of clear plastic, tape the plastic to the screen and then trace it. The didn't work either because my monitor horizontal to vertical ratio was different than the plotter. But I fixed that because my monitor has controls to adjust both. The next time, it worked fine. But it's a slow process. If all goes well, however, the results are rewarding. It strikes me particularly intriguing to do one of these, save the data to tape (or disk) and send it to a friend and let him watch it plot before his own eyes.

PROBLEMS

The PIC-PLOT program "trims" the two sides and bottom of the image. This can sometimes cause minor problems with the resultant plot.

The mechanism itself is delicate and must be treated with extreme care.

The ink in the nibs is watercolor, and dries out if the plotter is left unused for a period of time. They can be removed and recapped to prevent this problem

THE GOOD POINTS

It is without question the least expensive way to produce color hardcopy from a 2068.

Replacement nibs and paper are available from Radio Shack.

Overall I feel it's an excellent buy for the money. I love being able to produce color hardcopy, and these are the prices I came to Timex for in the first place.

WISH LIST

I wish I could plot a picture using PIC-PLOT and then load SCREEN-TO-PLOTTER and "touch-up" the picture I just plotted. Currently you can't do this since both the termination of PIC-PLOT and the initialization of SCREEN-TO-PLOTTER advance the paper feed on the 1520.

I wish the 1520 screen matched the 2068 screen so that it wouldn't "clip off" the edges.

I wish I could refill the pen nibs with other colors of ink.

I wish that when PIC-PLOT drew the horizontal lines that it started at the top of the image rather than at the bottom. This would allow me to see the image as it's being produced rather than waiting till it's done.

The hardware protion of this system is available in the Oliger tradition,i.e.,bareboard ($15), bareboard with parts ($21) and fully assembled ($30), but it's a very simple configuration with only 4 IC's much like an Oliger Parallel printer interface.

Incidentally, there are 200 copies of Timelinez this time around, each one has a different plot on it. So if you know anyone else who gets Timelinez, you can see a differnt example.

NEW FEATURE

John Discovered that the same interface can also be used to drive an OKIMATE 20 color printer. This printer is very much still in production, albeit slightly more expensive. I haven't gotten one of these, but the sample I saw really impressed me.

For more information, send a S.A.S.E. to John McMichael/1710 Palmer Dr/Laramie,WY 82070.

****

MC Pointers, Part Two, continued from last issue.

PROGRAM DESIGN

The design process breaks down into several stages. The design begins with a functional definition of the program.

Fine detail is ignored at this level and only a high level overview is defined

The following step is concerned with the decomposition of this overview into logical blocks, filling in the detailed functional requirements.

These blocks of functions are then defined as subprograms with an emphasis on compartimentalization and communication.

The subprograms are broken down to smaller logical modules, with emphasis on simple easy to understand function and structure.

The "Top Down" design is done entirely with symbolic names and provides an easy to read description of the whole program right down to the simple module level.

Now the process reverses and the functional modules are replaced with software modules. Then these modules are linked into more complex subprograms and the subprograms are finally linked together into the overal program. Note that the Assembler mnemonics come in use only when the program design has been reduced to the smallest practical module level. At that point the modules are written in M/L to impliment the functions and to interface with the system software and hardware. Modules must be able to communicate with each other through the use of flags, variables and the stack. A standard method will make the design simpler and easier to debug. Each module is designed interactively by coding and testing until the module is fully debugged. Then the module is added to the subprogram and the communication between modules is tested. Then the subprograms are tested as major functional blocks. And finally the subprograms are tested together as a system. At this time the user can provide feedback on the overall functional integrity and the userfriendliness of the Man/Machine interface. Remember that as a programmer you are familiar with the quirks of the machine but that no such assumption can be made about the user. Perhaps design for more than one level of user expertise. A simple menu driven interface for the novice and direct access for the more experienced user.

...cont'd on page 17.

The Variable Display Routine, Part Two, continued from last issue.

Some software hires programs use RET (instead of N/L and Interupt) to terminate each line and to return to a calling routine which processes the horizontal sync and display pointer updates. WRX16 uses a JP to terminate a dummy display line while the hires data file requires no line terminations at all.

After the last scan line execution of DFILE is interupted the B reg decrements to 0 and the Interupt routine returns to calling program at 28b. The routine continues with a CALL 292 which first saves the return address 28F in the IX register and then loads the number of blank lines below the screen from sysvar MARGIN. Next the routine restores all registers including A" and turns on the NMI generator before returning to the interupted BASIC or ML program.

So far so good but what about vertical frame synchronization?

Well the next time the A" reg overflows during a NMI instead of a JP (IX) to 281 it jumps to 28F which was the return address previously saved in IX ,followed by a JP 229 to the keyboard ,frame counter, and vertical sync service routines and finally ending at 27E with a CALL 292 which this time stores return address 281 in the IX register before loading the blank lines above the screen into A" and returning to the BASIC program.

Having come full circle,this completes the background description of the normal ZX video routines.

If we load IX with an address of an entirely different video routine we can do almost anything within the constraints of the hardware and video timing,from software hires to VDR.

Listing 1 begins with a short routine called INVOKE (INVK) to set the IX reg to point to the actual VDR routine. Note that we never execute the VDR routine directly.

The following routine restores the IX reg to point to the NORMAL (NORM) ZX display routine. This routine may not be required since during BASIC program development a simple keyboard procedure BREAK,SHIFT 0, ENTER will always return the NORMAL screen.

The next routine is VDR to which JP (IX) vectors. It consists of CALL VDP which takes the place of the routines at 281 and 285, followed by CALL 292 which exits to BASIC with the return address VDR1 in the IX register. When next interupted by an NMI which increments A" to overflow (ie last blank line), the CALL 220 is executed and the keyboard,vertical sync etc. gets serviced. Note that the registers are stacked first (220) so that the routine at 292 returns to VDR instead of the BASIC program and so the IX register can be restored to VDR.

Finally the jp 2A4 POPS the registers and returns to BASIC.

The Video Display Pointer (VDP) routine is used calculate the number of diplay lines and conversely the number of blank lines which are loaded into sysvar MARGIN. Then the location of the display (V-FILE) is calculated with respect to sysvar VARS and the offset in sysvar SEED. Then reg A is loaded with 100-DD characters per line (35) and a JP 41 starts the execution of the V-FILE via a part of the interupt routine which loads R,A ,Enables Interupt (EI) and JP (HL) to the actual start of V-FILE.

The next routine is VDR to which JP (IX) vectors. It consists of CALL VDP which takes the place of the ZX routines at 281 and 285, followed by CALL 292 which exits to BASIC with the return address VDR1 in the IX register. When next interupted by an NMI which increments A" to overflow (ie last blank line),JP (IX) vectors to VDR1 then CALL 220 is executed and the keyboard,vertical sync etc. gets serviced. Note that the registers are stacked first (220) so that the routine at 292 returns to VDR instead of the BASIC program and so the IX register can be restored to VDR.

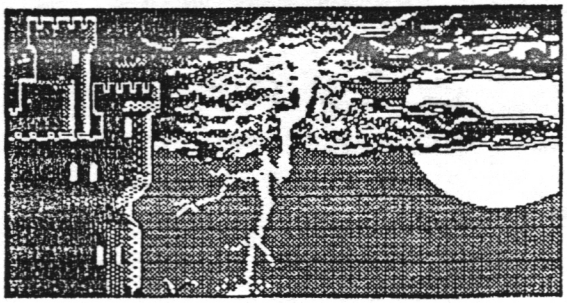
Finally the JP 2A4 POPS the registers and returns to BASIC.

The Video Display Pointer (VDP) routine is used calculate the number of diplay lines and conversely the number of blank lines which are loaded into sysvar MARGIN. Then the location of the display (V-FILE) is calculated with respect to sysvar VARS and the offset in sysvar SEED. Then reg A is loaded with 100-DD characters per line (35) and a JP 41 starts the execution of the V-FILE via a part of the interupt routine which loads R,A ,Enables Interupt (EI) and JP (HL) to the actual start of V-FILE.

```
        ;VDR1 (C) 1986 - W RIGTER

        ;VARIABLE DISPLAY ROUTINE

INVK
        LD IX,VDR;SET VDR VECTOR
        RET

NORM
        LD IX,281;SET NORM VECTOR
        RET

VDR
        CALL VDP;VAR DISPLAY PROC
        CALL 292;RETURN TO BASIC
        CALL 220;KEYBOARD/V SYNC
        LD IX,VDR;RESTORE IX PNTR
        JP 2A4;RETURN TO BASIC

VDP
        LD A,(4021);DISPLAY LINES
        AND 1F; 32 LINES MAXIMUM
        LD B,A;REG B IS THE CNTR
        RLCA; CHRLINES X 8=SCAN
        RLCA;
        RLCA;
        ADD A,21;CALC BLANK LINES
        CPL
        LD (4028),A;SYSVAR MARGIN
        LD C,8;8 SCAN / LINES
        LD HL,(4010);VAR TOP
        LD DE,(4032);SEED=OFFSET
        ADD HL,DE;TOP OF DISPLAY
        SET 7,H;ECHO ABOVE 32K
        LD A,DD;100-DD= CHR/LINE
        JP 41;EXECUTE DISPLAY
```

◇◇◇ A TOUCH OF CLAZX ◇◇◇

by F.Nachbaur



◇◇◇ A TOUCH OF CLAZX ◇◇◇

by F.Nachbaur

You may have noticed that last issue's BOLDFONT is like the native Sinclair character set, of the "sans-serif" type, i.e. it lacks the decorative strokes at the tops and bottoms of letters like 'L', 'T' and so on. So here is a character set that is, if you will, "avec-serif."

As this article will attest, the resulting font has a certain elegance that is not often seen in low-end computers, n'est ce pas?

Call the UDG definition routine with GOSUB 8000. See last issue for a simple program to display this, or any other, "customized" character set.

As before, the last 3 digits of each line # represents the CODE of the CHR$ being redefined. The inverse alphas are again lowercase; inverse symbols are again used for the missing symbols, & inverse numbers are arbitrarily used for superscripts.

By the way, your SRAM HI-RES application need not contain these [admittedly long] UDG definition subroutines. After RUNning these defining lines, you can enter or load other programs; your font will remain in your static RAM until you overwrite it.

For those of you who would like to play with these nifty {if I say so myself} fonts, but dread the thought of all that typing, contact VSUG for a tape.

NEXT TIME AROUND...

I'll send a rather unique, dast I say AMAZING, font that'll warm the cockles (whatever those are) of every true-blue Sinclair fancier's little heart. 'Till then,

CIAO FOR NOW...


```
7999 REM ELITE GOTHIC
8000 IF USR HR THEN LPRINT UDG;"
 ","00,00,00,00,00,00,00,00"
8010 IF USR HR THEN LPRINT UDG;"
"","00,6C,48,24,00,00,00,00""00,
36,12,24,00,00,00,00"
8012 IF USR HR THEN LPRINT UDG;"
£","00,24,7E,24,24,7E,24,00"
8013 IF USR HR THEN LPRINT UDG;"
$","00,10,38,50,38,14,38,10"
8014 IF USR HR THEN LPRINT UDG;"
:","00,00,00,18,00,00,18,00"
8015 IF USR HR THEN LPRINT UDG;"
?","00,3C,42,04,18,00,18,00"
8016 IF USR HR THEN LPRINT UDG;"
(","00,04,08,08,08,08,04,00"
8017 IF USR HR THEN LPRINT UDG;"
)","00,20,10,10,10,10,20,00"
8018 IF USR HR THEN LPRINT UDG;"
>","00,00,10,08,04,08,10,00"
8019 IF USR HR THEN LPRINT UDG;"
<","00,00,04,08,10,08,04,00"
8020 IF USR HR THEN LPRINT UDG;"
=";"00,00,00,3E,00,3E,00,00"
8021 IF USR HR THEN LPRINT UDG;"
+","00,00,10,10,7C,10,10,00"
8022 IF USR HR THEN LPRINT UDG;"
-","00,00,00,00,3E,00,00,00"
8023 IF USR HR THEN LPRINT UDG;"
*","00,00,28,10,7C,10,28,00"
8024 IF USR HR THEN LPRINT UDG;"
/","00,02,04,08,10,20,40,00"
8025 IF USR HR THEN LPRINT UDG;"
:","00,00,00,18,00,18,08,10"
8026 IF USR HR THEN LPRINT UDG;"
,","00,00,00,00,00,18,08,10"
8027 IF USR HR THEN LPRINT UDG;"
.","00,00,00,00,00,18,18,00"
8028 IF USR HR THEN LPRINT UDG;"
0","00,3C,46,4A,52,62,3C,00"
8029 IF USR HR THEN LPRINT UDG;"
1","00,10,30,50,10,10,7C,00"
8030 IF USR HR THEN LPRINT UDG;"
2","00,38,64,04,18,22,7E,00"
8031 IF USR HR THEN LPRINT UDG;"
3","00,3C,62,0C,02,62,3C,00"
8032 IF USR HR THEN LPRINT UDG;"
4","00,08,18,28,4A,FE,08,00"
8033 IF USR HR THEN LPRINT UDG;"
5","00,7E,40,7C,02,62,3C,00"
8034 IF USR HR THEN LPRINT UDG;"
6","00,18,10,20,7C,42,3C,00"
8035 IF USR HR THEN LPRINT UDG;"
7","00,7E,44,08,10,10,38,00"
8036 IF USR HR THEN LPRINT UDG;"
8","00,3C,42,3C,42,42,3C,00"
8037 IF USR HR THEN LPRINT UDG;"
9","00,3C,42,3E,04,08,18,00"
8038 IF USR HR THEN LPRINT UDG;"
A","00,18,24,42,7E,42,E7,00"
8039 IF USR HR THEN LPRINT UDG;"
B","00,7C,22,3C,22,22,7C,00"
8040 IF USR HR THEN LPRINT UDG;"
C","00,3C,46,40,40,46,3C,00"
8041 IF USR HR THEN LPRINT UDG;"
D","00,F8,44,42,42,44,F8,00"
8042 IF USR HR THEN LPRINT UDG;"
E","00,FE,42,78,40,42,FE,00"
8043 IF USR HR THEN LPRINT UDG;"
F","00,FE,42,70,40,40,E0,00"
8044 IF USR HR THEN LPRINT UDG;"
G","00,7C,86,80,9E,84,7C,00"
8045 IF USR HR THEN LPRINT UDG;"
H","00,EE,44,7C,44,44,EE,00"
8046 IF USR HR THEN LPRINT UDG;"
I","00,7C,10,10,10,10,7C,00"
8047 IF USR HR THEN LPRINT UDG;"
J","00,3E,04,04,C4,44,38,00"
8048 IF USR HR THEN LPRINT UDG;"
K","00,E6,48,70,50,48,E6,00"
8049 IF USR HR THEN LPRINT UDG;"
L","00,70,20,20,20,22,7E,00"
8050 IF USR HR THEN LPRINT UDG;"
M","00,C6,6C,54,44,44,EE,00"
```

```
"#$:?()><=+-*/;,.

0 1 2 3 4 5 6 7 8 9

ABCDEFGHIJKLMNOPQRSTUVWXYZ

!%&'!{}~_[@]←\↑'◇

0123456789

abcdefghijklmnopqrstuvwxyz
```

"The quick brown foxes jumped over the lazy dog."

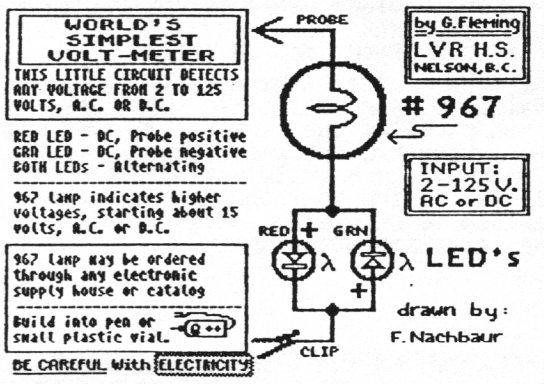
'THE QUICK BROWN FOXES JUMPED OVER THE LAZY DOG.'

$E=mc^2$ 2116=65536 $2H_2O \leftarrow 2H_2 + O_2$

$V \sim [\ SIN(t) * EXP(-k/t) + V_0\ ]$

```
8051 IF USR HR THEN LPRINT UDG;"
N","00,CE,64,54,4C,44,E6,00"
8052 IF USR HR THEN LPRINT UDG;"
O","00,7C,42,42,42,42,3E,00"
8053 IF USR HR THEN LPRINT UDG;"
P","00,FC,42,42,7C,40,F0,00"
8054 IF USR HR THEN LPRINT UDG;"
Q","00,7C,42,42,4A,46,3E,00"
8055 IF USR HR THEN LPRINT UDG;"
R","00,FC,42,7C,46,44,E7,00"
8056 IF USR HR THEN LPRINT UDG;"
S","00,3C,40,3C,02,62,3C,00"
8057 IF USR HR THEN LPRINT UDG;"
T","00,FE,92,10,10,10,38,00"
8058 IF USR HR THEN LPRINT UDG;"
U","00,EE,44,44,44,44,38,00"
8059 IF USR HR THEN LPRINT UDG;"
V","00,EE,44,44,44,28,10,00"
8060 IF USR HR THEN LPRINT UDG;"
W","00,EE,44,54,54,6C,44,00"
8061 IF USR HR THEN LPRINT UDG;"
X","00,EE,28,10,10,28,EE,00"
8062 IF USR HR THEN LPRINT UDG;"
Y","00,EE,44,28,10,10,38,00"
8063 IF USR HR THEN LPRINT UDG;"
Z","00,7E,44,08,12,22,7E,00"
8139 IF USR HR THEN LPRINT UDG;"
[","00,10,10,10,00,10,10,10"
8140 IF USR HR THEN LPRINT UDG;"
\","00,40,A4,48,10,24,4A,04"
8141 IF USR HR THEN LPRINT UDG;"
]","00,10,28,10,2A,44,3A,00"
8142 IF USR HR THEN LPRINT UDG;"
↑","00,18,10,08,00,00,00,00"
8143 IF USR HR THEN LPRINT UDG;"
←","00,10,10,10,10,00,10,00"
8144 IF USR HR THEN LPRINT UDG;"
£","00,0E,10,08,30,08,10,0E"
8145 IF USR HR THEN LPRINT UDG;"
©","00,70,08,10,0C,10,08,70"
8146 IF USR HR THEN LPRINT UDG;"
↑","00,00,00,60,92,0C,00,00"
8147 IF USR HR THEN LPRINT UDG;"
←","00,00,00,00,00,00,00,FE"
8148 IF USR HR THEN LPRINT UDG;"
£","00,0E,08,08,08,08,0E,00"
8149 IF USR HR THEN LPRINT UDG;"
©","00,38,44,58,5A,42,3C,00"
8150 IF USR HR THEN LPRINT UDG;"
↑","00,70,10,10,10,10,70,00"
8151 IF USR HR THEN LPRINT UDG;"
←","00,10,20,40,FE,40,20,10"
8152 IF USR HR THEN LPRINT UDG;"
£","00,C0,60,30,18,0C,06,00"
8153 IF USR HR THEN LPRINT UDG;"
©","00,10,38,54,10,10,10,00"
8154 IF USR HR THEN LPRINT UDG;"
↑","00,18,08,10,00,00,00,00"
8155 IF USR HR THEN LPRINT UDG;"
←","00,10,28,44,82,44,28,10"
8156 IF USR HR THEN LPRINT UDG;"
£","1C,2C,24,34,38,00,00,00"
8157 IF USR HR THEN LPRINT UDG;"
©","08,18,08,08,1C,00,00,00"
8158 IF USR HR THEN LPRINT UDG;"
```

```
↑","18,24,08,10,3C,00,00,00"
8159 IF USR HR THEN LPRINT UDG;"
←","38,04,18,04,38,00,00,00"
8160 IF USR HR THEN LPRINT UDG;"
£","20,28,3C,08,08,00,00,00"
8161 IF USR HR THEN LPRINT UDG;"
©","3C,20,38,04,38,00,00,00"
8162 IF USR HR THEN LPRINT UDG;"
↑","18,20,38,24,18,00,00,00"
8163 IF USR HR THEN LPRINT UDG;"
←","3C,24,08,10,10,00,00,00"
8164 IF USR HR THEN LPRINT UDG;"
£","18,24,18,24,18,00,00,00"
8165 IF USR HR THEN LPRINT UDG;"
©","18,24,1C,04,18,00,00,00"
8166 IF USR HR THEN LPRINT UDG;"
↑","00,00,00,3E,42,42,3F,00"
8167 IF USR HR THEN LPRINT UDG;"
←","00,60,20,3C,22,22,7C,00"
8168 IF USR HR THEN LPRINT UDG;"
£","00,00,00,3C,46,40,3C,00"
8169 IF USR HR THEN LPRINT UDG;"
©","00,0E,04,3C,44,44,3E,00"
8170 IF USR HR THEN LPRINT UDG;"
↑","00,00,00,24,7E,40,3C,00"
8171 IF USR HR THEN LPRINT UDG;"
←","00,1C,26,20,78,20,70,00"
8172 IF USR HR THEN LPRINT UDG;"
£","00,00,00,3E,44,3C,84,78"
8173 IF USR HR THEN LPRINT UDG;"
©","00,C0,40,78,44,44,EE,00"
8174 IF USR HR THEN LPRINT UDG;"
↑","00,18,00,38,08,08,3E,00"
8175 IF USR HR THEN LPRINT UDG;"
←","00,18,00,38,08,08,48,30"
8176 IF USR HR THEN LPRINT UDG;"
£","00,C0,48,50,70,48,E6,00"
8177 IF USR HR THEN LPRINT UDG;"
©","00,70,10,10,10,10,7C,00"
8178 IF USR HR THEN LPRINT UDG;"
↑","00,00,00,F8,54,54,56,00"
8179 IF USR HR THEN LPRINT UDG;"
←","00,00,00,F8,44,44,E6,00"
8180 IF USR HR THEN LPRINT UDG;"
£","00,00,00,38,44,44,38,00"
8181 IF USR HR THEN LPRINT UDG;"
©","00,00,00,FC,42,7C,40,E0"
8182 IF USR HR THEN LPRINT UDG;"
↑","00,00,00,3E,44,3C,04,03"
8183 IF USR HR THEN LPRINT UDG;"
←","00,00,00,6C,36,20,70,00"
8184 IF USR HR THEN LPRINT UDG;"
£","00,00,00,3C,30,0C,3C,00"
8185 IF USR HR THEN LPRINT UDG;"
©","00,20,20,F8,20,26,1C,00"
8186 IF USR HR THEN LPRINT UDG;"
↑","00,00,00,CE,44,44,3E,00"
8187 IF USR HR THEN LPRINT UDG;"
←","00,00,00,E6,44,28,10,00"
8188 IF USR HR THEN LPRINT UDG;"
£","00,00,00,C6,54,7C,44,00"
8189 IF USR HR THEN LPRINT UDG;"
©","00,00,00,66,18,18,66,00"
8190 IF USR HR THEN LPRINT UDG;"
↑","00,00,00,CE,44,3C,04,38"
8191 IF USR HR THEN LPRINT UDG;"
←","00,00,00,7E,48,12,7E,00"
8200 RETURN
```



The ZX/TS True 64K Internal NVM Upgrade

by V. Lee

Ever since I first heard of the Hunter Board, I wanted to expand the memory on my ZX81 and on my TS1000. The Hunter Board allowed 8K of Battery Back Up RAM to store utilities like SDS or Mini-Xmod in the ZX/TS 8-16K region. The programs would remain in the Non-Volatile Memory even after the computer was turned off, eliminating the chore of reloading the same programs. And being RAM, the old programs were easily changed by placing a new program into the same area.

I began experimenting with 8K Static Ram Chips until I saw an article written by Tim Stoddard. He showed an ingenious method of decoding two 32K SRAM chips to add the full 64K of memory internally. Say goodbye to the old Rampack Wobble.

If a couple of batteries were also added to his circuit, we could have a 64K version of the Hunter Board. Almost all programs will run in the normal 16-32K work space, leaving the memory outside of this area available for storing a variety of different programs. This outside area is not affected by "NEW" as long as RAMTOP is not raised.

The 8-16K area is still used for running SDS or WRX16 HIRES software, but the 32-48K area is now used for running resident programs like Memotext, Hi-Z, ZXTerm-80 and others which have been designed to be relocated. The area above 48K is used as a Ramdisk since machine code will not run in this area.

There are various machine code routines which will allow a program from the work area to be saved to RAM. This use of RAM as a "diskdrive" is known as a Ramdisk. I've made some attempts on a Ramdisk program in the April and September 87 issues of ZX-Appeal. But the best example is an Operating System written for the "Delta Device", a 32K NVM system. This program provides hotkey function, internal and transient commands as well as a directory of the stored "files".

I didn't feel there was enough room available for relocated programs. Hi-Z alone occupies the entire 32-48K block. There were two options, buy another 32K SRAM chip for Bank Switching or come up with a gimmick to increase the memory. I decided to choose the route which costs the least amount of money.

There just happens to be 8K of available RAM that is hidden because of the ROM. I decided to connect this block to the second 8K block of RAM and then move them to a new location by reconfiguring the decoding. This new 16K block which I call the "Auxiliary Memory" can be switched in when needed to the 32-48K region. It's great for hiding programs like Hi-Z or for raising Ramtop up to 48K without disturbing the main memory.

The old 40-48K block is made to travel incognito to fill the void. It can be accessed as the 40-48K block or as the 8-16K block, making this area extremely flexible. Data placed by one Address can be read or written over by the other. There is a 32768 Byte difference between the two Addresses.

Remember not to place HIRES software in the old 40-48K block as it will try to overwrite itself. This does not affect HIRES software placed in the Auxiliary Memory.

P A R T S L I S T

U1	74LS10N, 3-INPUT NAND GATE
U2	74LS08N, 2-INPUT AND GATE
U3	74LS145N, BCD TO DEC. DRIVER
U4,5	HM62256LP-15, 32KX8 LOW PWR SRAM
D1,2	1N34A, GERMANIUM DIODE
C1	0.1uF, 100V MONO CERAMIC CAP.
C2	2.2uF or 3.3uF, 16V TANTALUM CAP.
R1,2,3	4.7K, 1/4W RESISTOR
R4	10K, 1/4W RESISTOR
PB1	PUSHBUTTON SWITCH N/O
SW1	TOGGLE SWITCH DPDT
SW2	TOGGLE SWITCH SPST
BATT	3v, 2X1.5V AAA ALKALINE BATTERY WITH HOLDER

Assembling the 64K NVM

If you are thinking about building this circuit, you should first examine your ZX/TS system to see if it is compatible. Hardware like Disk Drives and Byte-Back Modems require certain blocks of memory to be disabled. This upgrade fills the entire memory map and would require alterations in the decoding.

You might also prefer making a circuit board to construct this project. Almost all of the signals are available from the card edge connector at the back of the computer. This circuit was assembled with point-to-point wiring. It allowed quick alterations and allowed the circuit to fit inside the computer. 28 gauge ribbon cable was used for the switch and for the power connections while 30 gauge wire wrap wire was used for the rest. The assembling directions in the following paragraphs are on the point-to-point method.

Between these two extremes is a third method of modifing an existing memory board to fit the circuit. There are various electronic outlets and computer swap meets which sell used computer boards for well under five dollars. Some of these boards have come out of video arcade games. Look for a board which has 28 pin sockets that are "bussed". This allows a more acceptable method of connecting the 32K SRAM.

The 32K SRAM chips are CMOS devices and are static sensitive. Use a grounded soldering iron and ground yourself by touching a large metal object before handling the ICs.

The pins on the three decoding chips were straightened to allow the chips to be glued to the top of the Z80. From here the required signals were easily tapped. Pins 10, 12 and 13 of the ULA chip are bent away from the socket. This allows a connection for NOT M1 to pin 10 and removes the ULA from any memory selecting. The CSROM signal from the new decoder is applied to the far end of R28. This allows external devices like Printer Drivers to deselect the ROM. On the TS board, resistor R28 is located beside the ULA chip while on the ZX board, it is located below.

The internal RAM could have been decoded as part of this circuit, but I decided to just replace it with a 28 pin socket. On the ZX board, check the right side of the RAM socket to make sure that a jumper is connected to L2 and not to L1. This switchs the signal from +5V to A10.

The two 32K SRAM chips are mounted on top of each other. All of the pins from one chip are soldered to the corresponding pins on the other chip except for pins 20 and 27 which are CS and WE. All of the pins fit into the RAM socket except for pin 28 which is connected to the Battery Back Up circuitry, pins 1, 2, 23 and 26 which are connected to the Address lines taken from the cathode side of the keyboard diodes and pins 20, 22 and 27 which are connected to the control lines from the Decoding Section. The bottom left pin of the 32K SRAM which is pin 14, fits into the bottom right pin of the socket. This lines up all of the signals from the socket to the proper pins on the SRAM.

The Battery Back Up circuitry consists of two Alkaline AA batteries connected in series to produce the required 3V. Make sure that the polarity of the batteries, diodes and capacitor are facing the correct direction.

A surplus TI keyboard was also connected to the system. Some of the unused keys were used as part of this circuit. The "Alpha Lock" key was used for the Write Protect and the "Ctrl" and "Fctn" keys were connected in series to form the RESET switch.

I did not include any automatic Write Protect circuitry in the design since I usually run the system with the memory in the Write Protect mode. This protects the data from accidental crashes. Before shutting down the system, make sure that the Memory Mode switch has been turned to Main Memory to protect the Auxiliary Memory. And make sure that the Write switch has been turned to Write Protect to protect the Main Memory. This prevents the data from being corrupted by the system when it turns on or off.

Troubleshooting Hints

Once the circuit has been completed, you should be able to move RAMTOP up to 64K. However if the ROM has not been selected, the screen will just flash when the ZX/TS is powered up. If the RAM has not been selected, the screen will be blank without the inverse K cursor appearing. And if the chips are incorrectly selected, the system will crash. Recheck the wiring to the decoders and to the memory chips and examine the control lines.

Turn off the Write Protect switch and use the commands;

```
POKE 16388,255
POKE 16389,255
NEW
```

to raise RAMTOP to 64K. Now use the commands, PRINT INT ((PEEK 16388+256*PEEK 16389+1)/1024), and the value 64 should appear. Press the RESET switch and then switch in the Auxiliary Memory. Leave the WRITE Protect switch in the off position. Again you should be able to move RAMTOP up to 64K. If the returned values are less than 64, the difference indicates where the error has occured in the decoding. Recheck the Address line connections.

Next time we'll start from the beginning to design your own decoding schemes and find some uses for the available gates.

References

Hunter Board
by Paul Hunter
Radio Electronics, Jul 83
Radio Electronics, Aug 83

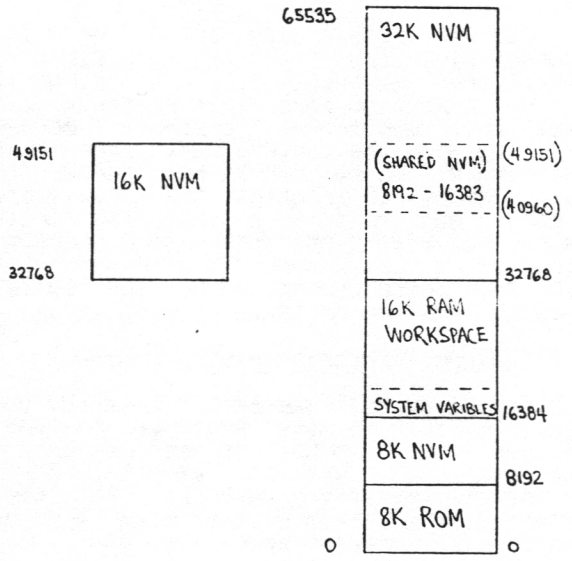
Internal 64K RAM for the TS/ZX
by Tim Stoddard
Time Designs Vol. 3#4, May/Jun 87

Run TS1000 Machine Code in High Memory
by John Oliger
SyncWare News Vol. 2#5, May/Jun 85

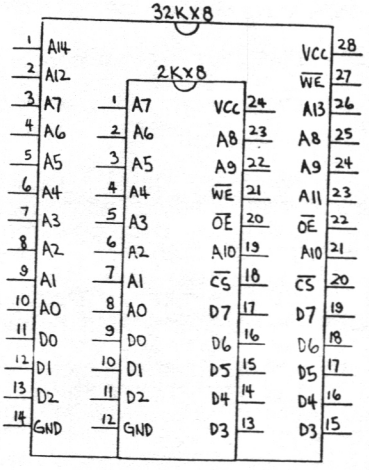
Built-In NVM
by Gerd Breunung
ZX-Appeal, Jul/Aug 86
SyncWare News Vol. 4#1, Sep/Oct 86

Other points of interests:

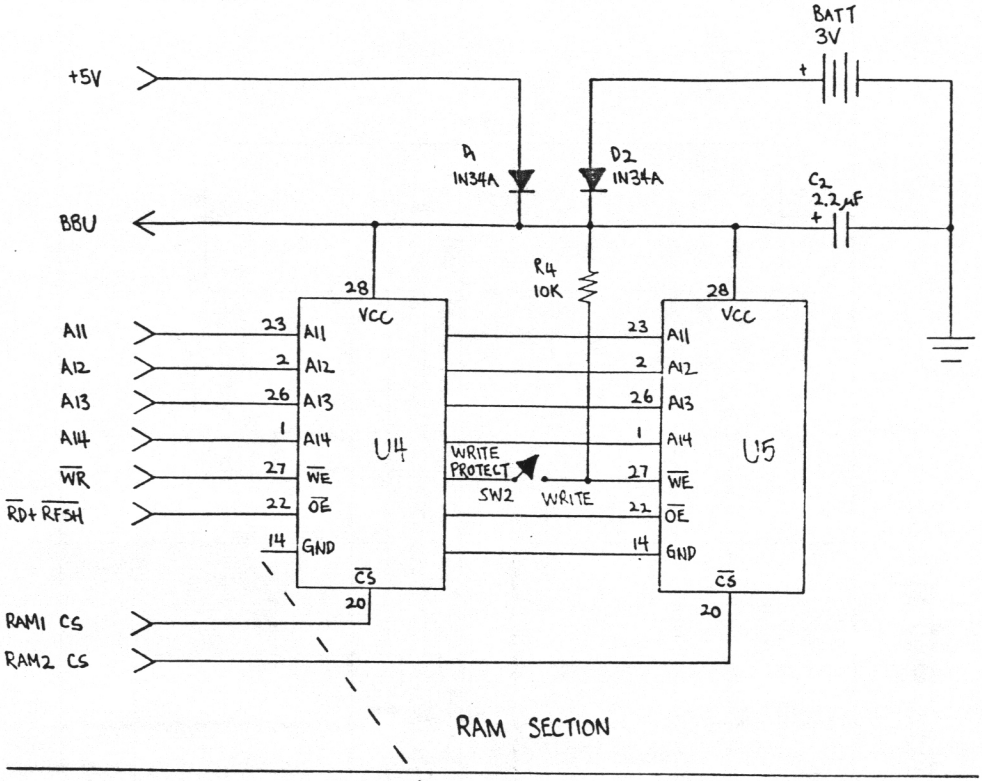
24K NVM, ZX-Appeal, Apr 88
32K NVM Delta Device, ZX-Appeal, Dec 87
ZX-Appeal, Sum 88

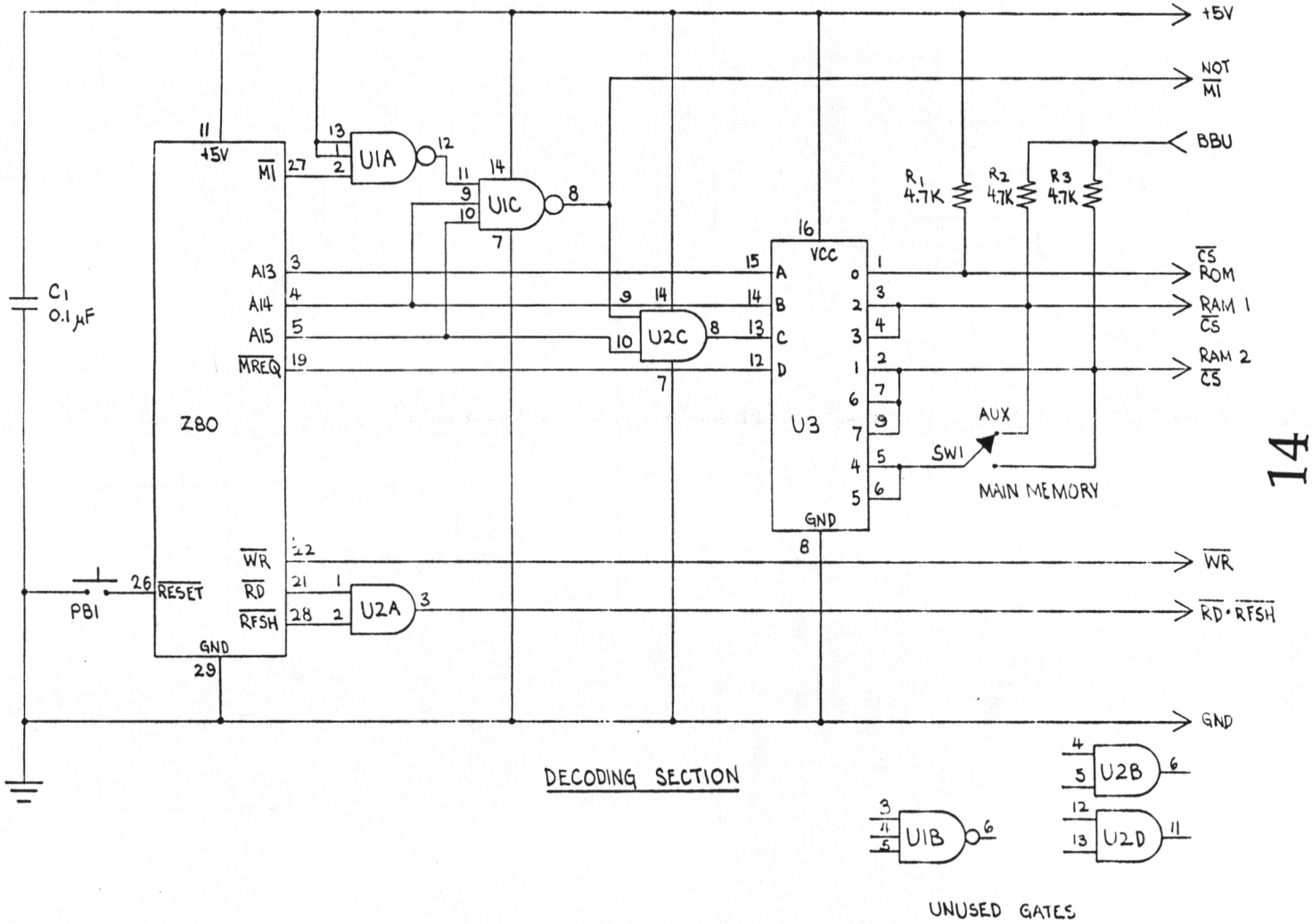


ZX/TS TRUE 64K NVM MEMORY MAP



COMPARISON OF THE TS 2KX8 STATIC RAM WITH THE UPGRADE 32KX8 STATIC RAM


BATT
3V
+5V
D1
IN34A
D2
IN34A
C2
2.2μF
B8U
R4
10K
28
VCC
U4
U5
A11
A12
A13
A14
WR
RD+RFSH
WE
OE
GND
CS
WRITE
PROTECT
SW2
WRITE
RAM1 CS
RAM2 CS
RAM SECTION
NOT
MI
CS
ROM
ULA
MI
RAMCS
ROMCS
BEND PINS
AWAY FROM
SOCKET
R28
ZX81 ONLY
Z80
RAM
ROM
L1
L2
TS1000 BOARD


+5V
NOT M1
BBU
C1
0.1 µF
11
+5V
M1
27
A13
3
A14
4
A15
5
MREQ
19
Z80
WR
22
RD
21
RFSH
28
RESET
26
PB1
GND
29
U1A
13
2
12
U1C
11
9
10
14
8
7
U2A
1
3
U2C
9
14
10
8
7
U3
16
VCC
15
A
14
B
13
C
12
D
1
0
3
2
2
1
7
6
9
7
5
4
6
5
GND
8
R1
4.7K
R2
4.7K
R3
4.7K
CS ROM
RAM 1 CS
RAM 2 CS
AUX
SW1
MAIN MEMORY
WR
RD · RFSH
GND
DECODING SECTION
U1B
3
1
5
6
U2B
4
5
6
U2D
12
13
11
UNUSED GATES

A problem common to QL owners is one of overheating...their machines, that is. Harvey T. sends along a reprint from QUANTA containing a fix for this problem.

A CURE FOR OVERHEATING

I see that there are still people with overheating problems (from the contents of the August Quanta). I would like to address this problem and hopefully PUT IT TO REST!

First, overheating is a symptom of the problem and not the problem. The QL uses a 1 amp 7805 voltage regulator. Everyone knows that this device gets very hot. With the lid open, you just can't touch the chip without burning your finger. However, you can touch the edge of the rather large heatsink. Hmmm, it's cold isn't it (relatively speaking)?

The 7805 chip requires bypass capacitance on the input (9 Volt side) AT the regulator and not some distance away (as the QL and ZX81 have). The 7805 has built in thermal protection so that if it gets too hot it will shut off. A locked up QL is not due to overheating. A QL with a black screen is an overheating problem.

If you check the specs for the 7805, you will see that its power output must be derated as the temperature of the part goes up.

Now let's discuss the real problem.

With improper bypassing and when the QL warms up, any electrostatic discharge into the air (from the monitor or walk by) can cause the 7805 to deregulate or oscillate. This surge enters by way of the power cable. Solution? Add a 0.1 microfarad capacitor (type 104, 16 working volts or greater) from the input to ground (right side to center pin when installed). Solder it right to the 7805 pins. Don't use a large capacitor to level the voltage on the input or you may blow your Microdrives (learned from experience). You might also consider routing your power cable away from your monitor and your feet.

It is interesting that the heat sink doesn't get very hot on the edges. Well it looks like a job for heatsink compound. Use a lot of it and don't forget to put some under the 7805. Don't skimp like Sinclair did. Now you'll find that the whole heatsink gets hot.

Better yet, add another 7805 to the heatsink. Cut the leads off about half way and screw it down on top of the first 7805, but point it the other way. Solder jumper wires criss cross back to the first 7805 and tie the two middle pins together. Put a flat washer under the new 7805 to take up the slack and fill it in with heat sink compound. You may have to bend up the regulator on the MDV to be sure that it doesn't contact the added 7805.

Consider also adding a bypass diode from 5 volts to 9 volts (banded end towards the 9 volt pin) on one of the regulators to prevent the 5 volts from ever rising above the 9 volt line.

I would suspect that someone with a regulator oscillation problem (previously refered to as overheating) would have other symptoms crop up, such as, blown ROMs and such. ROMs are 5800 times more likely to blow than a 74LS00, which has an estimated life of 1000 years (from reliability studies). I wonder how many QLs went back to the shop with blown ROMs within three months of purchase? I'll wager that quite a few did. ROMs simply cannot take any oscillation on the power grid.

One last point. Now that we are dissipating heat more efficiently, the case will get considerably warmer over the heatsink. I have also used large stick on feet to prop up the back of the QL so that it is about 1 1/4" from the table. I added small feet in from to allow air to pass under the front of the QL. This will allow air to flow in under the Microdrives and out the back more efficiently. (I find the increased angle easier for typing too.)

Consider this a power upgrade. You now have 2 AMPS of current. You can also get rid of all those silly noisy fans that you have blowing dust all over the inside of your QL.

Hardware Review

PSION computer

Imagine if you will, a hand held calculator. It looks like any you have seen but with a few more buttons. Now imagine this–

- A complete and easy to use data base program for storage and retrieval of any information,
- An Appointment calendar with time slots every half hour until 1999,
- A full function calculator with 10 memories and programming abilities,
- An alarm clock with 8 alarms available every 24 hours,
- A built in programming language as easy to use as BASIC,
- A Lotus 1-2-3 compatible spreadsheet,
- A typing tutor to teach you how to achieve 40 - 50 wpm. on the keypad,
- A 5 language translator to and from English, French, Spanish, German and Italian,
- A celestial navigator with the location of 27 stars, the moon and the sun to determine position and course at sea,
- A bar code reader and laser scanner interface to operate as a point of sale cash register and run inventory control packages,
- A serial port to connect to modems, printers, IBM and Macintosh computers, exchanging files,
- A battery operated 80 column printer slightly larger than the unit itself,
- A time billing program for Doctors and Lawyers, a route management program for milkmen and other delivery routes,
- A total of 3 interchangeable "disk drives" with RAM and ROM packs of up to 128K for an available storage of 284K at one time
- A development package with most of the combined features of Lotus 1-2-3 and DBase,
- A quickly increasing collection of software to accomplish almost any task,
- A price starting at well under $500!

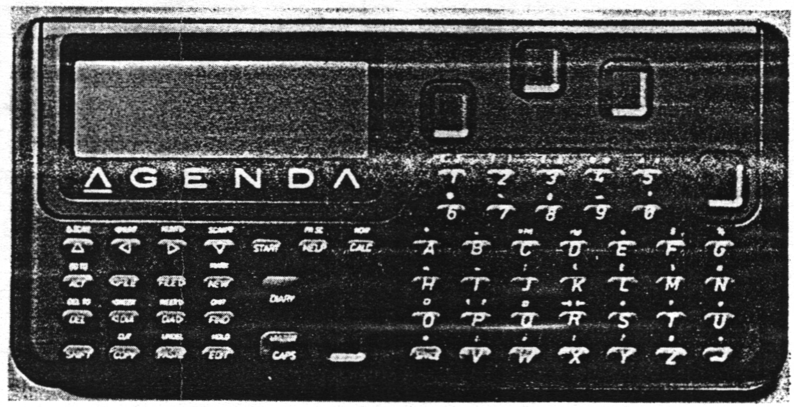
Manufactured for the last 2 years in England and now available in North America. PSION will soon be a very recognized name to readers of this publication. It truly is on the "Cutting Edge".

****

SMALLER THAN A LAPTOP— HERE'S THE POCKET COMPUTER

In the drive to take computing the next logical step down from laptops, Microwriter Systems Ltd. of Mitcham in Surrey, England, has used every trick in the book. The company's new pocket-size computer, called Agenda, uses surface-mount technology throughout. An application-specific integrated circuit forms the glue logic around its Hitachi 6303 processor and its 320-Kbyte internal memory. Battery-backed, removable random-access-memory cards make up the data-storage unit. The result is a computer that measures just 175 by 85 by 18 mm—the same size as a pocket diary.

The innovation Microwriter is counting



Agenda is a full-fledged computer with word-processing, data-base, and diary functions built into a package the size of a pocket diary.

on to give Agenda an edge over its competitors, though, is a specially developed keyboard known as the Chord keyboard. It occupies an area about 3.5 in. by 4 in. and consists of just five keys, one for each fingertip, arranged in an arc.

Each letter of the alphabet is formed by pressing different combinations of these five keys, says Microwriter managing director Mike Davies. "Our biggest problem was developing a mechanism for the keyboard that provides the correct tactile response with pressures of less than 20 grams," he says. As with Morse code, the keyboard is configured so the most frequently used characters are formed by hitting the simplest combinations of keys. The letter "E," for example, is made simply by pushing the forefinger's key alone, and the letter "S" by pushing the third finger's key alone. The letter "Z," however, requires hitting the keys for the thumb, middle, and little fingers simultaneously. "It sounds complicated," Davies says, "but most people become proficient in less than 20 minutes, and an accomplished user can type with one hand at around 50 words a minute."

The machine also has a conventional, albeit very small, alphanumeric keyboard. Another 30 buttons select functions from the machine's built-in word-processing, data-base, and diary software.

The machine uses a liquid-crystal display showing four lines of 20 characters. Power is from rechargeable NiCad batteries.

Internal software is held in a 32-Kbyte read-only memory. Third-party software on RAM cards will be available shortly, Davies says. The first externally loadable software will be a Basic compiler and an XModem communications protocol converter, to allow the interchange of binary files with other machines.

Compatibility with desktop computers and printers has been provided for with both RS-232 serial and Centronics parallel data ports.

The machine is listed in the UK at $250. Additional 32-Kbyte RAM cards cost around $50 each.

****

...cont'd from page 6.

DESIGN EXAMPLE

It is time to come down to earth and make a start on some practical ideas. The first step is to aquire some tools for program development. The most important of these is a Z80 assembler. Next comes the FILEHANDLER with the idea of creating a library of modules which have universal application.

The system software and user accesable subroutines form an existing library and we will use and discuss them at various times. Fully annotated system documentation is ofcourse important to interface with the system software so dissassemble and anotate or acquire existing docs from someone else.

The program presented in listing 1 is RAMDOS1000 which will be used to organize our library of modules and other software. It is written in a special way to allow it's use with various system organization and capacity , in fact it can be located anywhere in memory. Examine this program and see the benefits and special requirements of writing software in Relocatable Machine Code. The program listing is fully annotated but to describe it's applications requires a 38 page manual which can not be covered in this short article. For more information on RAMDOS1000 write the author . [4]

RELOCATABLE M/C SUMMARY

I will end with a summary of the what,where,when,how of relocatable code.

1.Relocatable code (RC) is the product of a M/L program that does not use absolute memory addresses when referring to itself. In Z80 programs for example, never use CALL NN, JP NN, LD RR,(NN) where NN refers to a part of the RC itself. Absolute memory references to the ROM or SYSTEM VARIABLES however are allowed.

2.RC by definition can be located anywhere in memory [3], but is usually found in the 8K-16K area or can be embedded in any BASIC REM statement.

3.RC should be used whenever programs have to coexist in memory or when software is developed in small modules and linked at run time with other programs.

4.The important instructions mentioned in 1. must be replaced with equivalents, albeit at the cost of more complex code. Several examples will serve as recipes for your own program development.

References
---

[1] Standard Dictionary of Computers and Information Processing.
[2] Websters New Colligiate Dictionary.
[3] In the ZX81 no machine code can be executed above 32K.
[4] RAMDOS1000 is available as part of the NVMS system.

```
  10 REM   LOVE LETTER
  20 REM
  30 REM   TS1000 OR 2068
 100 REM
 110 PRINT "DO YOU HAVE A GIRLFR
IEND (G)OR A BOYFRIEND(B)"
 120 INPUT G$
 130 PRINT "WHAT IS YOUR FIRST N
AME?"
 140 INPUT N$
 150 IF G$="G" THEN GOTO 400
 200 PRINT "WHAT IS YOUR BOYFRIE
ND'S NAME?"
 210 INPUT F$
 220 PRINT "WHAT COLOR IS HIS HA
IR?"
 230 INPUT H$
 240 PRINT "WHAT COLOR ARE HIS E
YES?"
 250 INPUT E$
 260 PRINT "HIS BEST FEATURE IS
HIS (WHAT)?"
 270 INPUT B$
 280 PRINT "HOW LONG AGO DID YOU
 FIRST MEET HIM?"
 290 INPUT D$
 300 PRINT "WHERE WAS THAT?"
 310 INPUT W$
 320 PRINT "WHAT IS ONE THING YO
U LIKE TO DO TOGETHER?"
 330 INPUT T$
 340 PRINT "WHEN DID YOU SEE HIM
 LAST?"
 350 INPUT L$
 360 GOTO 600
 400 PRINT "WHAT IS YOUR GIRLFRI
END'S NAME?"
 410 INPUT F$
 420 PRINT "WHAT COLOUR IS HER H
AIR?"
 430 INPUT H$
 440 PRINT "WHAT COLOR ARE HER E
YES?"
 450 INPUT E$
 460 PRINT "HER BEST FEATURE IS
HER (WHAT)?"
 470 INPUT B$
 480 PRINT "HOW LONG AGO DID YOU
 FIRST MEET HER?"
 490 INPUT D$
 500 PRINT "WHERE WAS THAT?"
 510 INPUT W$
 520 PRINT "WHAT IS ONE THING YO
U LIKE TO DO TOGETHER?"
 530 INPUT T$
 540 PRINT "WHEN DID YOU SEE HER
 LAST?"
 550 INPUT L$
 600 CLS
 610 PRINT "MY DEAREST ";F$;","
 620 PRINT
 630 PRINT "I HAVE MISSED YOU SO
 MUCH SINCE I SAW YOU ";L$;"."
 640 PRINT "I LOVE THE LOOK OF Y
OR ";H$;" HAIR."
 650 PRINT "I LOVE TO GAZE INTO
YOUR BEAUTIFUL ";E$;" EYES."
 660 PRINT "BUT MOST OF ALL I LO
VE YOUR ";B$;"."
 670 PRINT "MY LUCKIEST DAY WAS
THE DAY I MET YOU ";W$;"."
 680 PRINT "DO YOU REMEMBER? --
IT WAS ";D$;" AGO."
 690 PRINT "TONIGHT LETS GET TOG
ETHER AND ";T$;"."
 700 PRINT "UNTIL THEN, ALL MY L
OVE,"
 710 PRINT
 720 PRINT "                ";N$
```

Reprinted from the Jul-Aug/87 issue of TIMELINZ

ALARM CLOCK by Kevin Leung

This is a program that will get you up in the morning, or any other time. The program is written in basic so it should be easy to follow. If you do not want the alarm feature, just put (0) for the alarm time. If you are a heavy sleeper, just insert the mic plug into the ear jack and then you can set the sound level. (If you want to speed up or slow down the clock, change line 101 to Pause 75, 125, etc.). The program has an auto-run feature once it has been loaded.
NOTE: This program is for the 2068 only.....

```
   1 PRINT "bell hr?"
   2 INPUT ah
   3 CLS
   4 PRINT "min bell?"
   5 INPUT am
   9 CLS
  10 PRINT "hr?"
  20 INPUT h
  25 IF h>12 THEN GO TO 9
  30 CLS
  40 PRINT "min?"
  50 INPUT m
  55 IF m>60 THEN GO TO 30
  56 CLS
  60 PRINT "sec?"
  63 GO TO 80
  70 INPUT s
  75 IF s>60 THEN GO TO 56
  80 CLS
  90 PRINT AT 10,10;h;"  ";m;"
";s
 100 LET s=s+1
 101 PAUSE 100
 105 IF s<>61 THEN GO TO 80
 106 IF s=61 THEN LET s=0
 110 IF s=61 THEN GO TO 120: GO
TO 80
 120 LET m=m+1
 125 IF ah=h AND am=m THEN GO TO
 9000
 130 IF m=61 THEN LET m=0
 140 IF m=60 THEN GO TO 160
 150 GO TO 80
 160 LET h=h+1
 161 IF h=13 THEN LET h=1
 162 IF ah=h AND am=m THEN GO TO
 9000
 163 GO TO 80
9000 BEEP 10,50
9500 LET s=s+50
9900 GO TO 80
9990 STOP
9995 SAVE "al clock" LINE 1
9999>REM By Kevin Leung
```

CAPITAL AREA

TIMEX SINCLAIR USERS GROUP

C. A. T. S.

DEAR TIMEX/SINCLAIR ENTHUSIAST:

WE WOULD LIKE TO TAKE THIS OPPORTUNITY TO INVITE YOU TO ATTEND THE UPCOMING CAPITAL AREA TIMEX/SINCLAIR CAPITALFEST ON MAY 6 AND MAY 7, 1989. BANQUET FRIDAY NIGHT, MAY 5, 1989.

FEST TO BE HELD AT THE HOWARD JOHNSON INN, ROUTE 450 AND THE BELTWAY (EAST-SIDE), NEW CARROLLTON, MD.

HOWARD JOHNSON'S IS CONVENIENTLY LOCATED FOR EASY ACCESS BY ROAD, SUBWAY (NEW CARROLLTON, MD. STOP), AMTRAK FROM NEW YORK AND BOSTON TO THE NORTH AND ALL POINTS SOUTH, AND BY AIR INTO WASHINGTON NATIONAL AND BALTIMORE — WASHINGTON INTERNATIONAL.

CATS, OUR USERS GROUP, CURRENTLY WITH OVER 100 MEMBERS WILL BE HOSTING THIS AFFAIR.

SHARP'S, ZEBRA, AND MANY OTHER VENDORS WILL BE SPONSORING AND ATTENDING.

SEMINARS, DOOR PRIZES, AND MANY SURPRISES AWAIT YOU.

THIS CAPITALFEST WILL BE FULLY ADVERTISED WITH THOUSANDS OF FLYERS BEING SENT ALL OVER THE COUNTRY.

HOTEL ROOM RATE IS $62.00 A NIGHT ALONG WITH ONE FREE ADMISSION TO SHOW.

BANQUET, FRIDAY NIGHT WILL RUN AROUND $17.00.

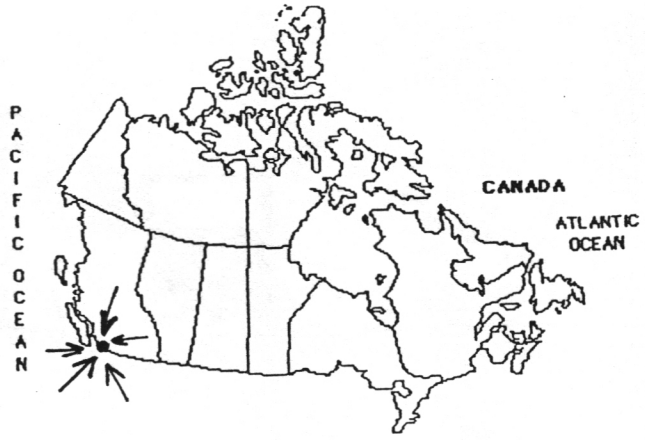
TABLES WILL BE $25.00 EACH.

IF YOU ARE INTERESTED IN MORE INFORMATION, PLEASE LET US KNOW.

GET THE DETAILS!!!

LOVE TO HAVE YOU AT THE FEST.

C.A.T.S. CAPITALFEST
P. O. BOX 24
GARRETT PARK, MARYLAND 20896-0024
301-439-8756



VSUG

The Vancouver Sinclair Users Group has been in existence since 1982. We are a support group for the owners and users of all SINCLAIR and TIMEX computers.

Pres:- Gerd Breunung PH#(604) 931-5509
V/Pres:- Glenn Read
Sec:- Harvey Taylor
Treas. & N/L Editor:- Rod Humphreys

Our membership dues are only $15.00/year and may be sent to the Treasurer:

Rod Humphreys
2006 Highview Place
Port Moody, B.C., V3H 1N5

Members of VSUG receive a monthly issue of ZXAppeal - our newsletter.

ZXAppeal accepts advertising. Our **PREPAID** rates are:

$10.00 – full page
$8.00 – 1/2 page
$5.00 – 1/4 page

ZXAppeal is distributed to approx 30 other SINCLAIR User Groups throughout North America as well as overseas via the NETWORK. NETWORK correspondence may be directed to the Editor at the above address.